Scientific Reports

https://doi.org/10.1038/s41598-026-58099-5

Article in Press

Battery performance prediction using supervised learning: a comparative study

Received: 18 November 2025

Accepted: 11 June 2026

Published online: 01 August 2026

Cite this article as: Reddy M.S. & Monisha M. Battery performance prediction using supervised learning: a comparative study. *Sci Rep* (2026). https://doi.org/10.1038/s41598-026-58099-5

Mandeddu Sudhakar Reddy & M. Monisha

Battery Performance Prediction Using Supervised Learning: A Comparative Study

***Mandeddu Sudhakar Reddy, M. Monisha**

Department of Electronics and Communication Engineering, Vels Institute of Science Technology and Advanced Studies, Chennai, India

sudhakarreddymandeddu@gmail.com, monisha22@gmail.com

***Corresponding E-mail -** sudhakarreddymandeddu@gmail.com

Abstract

Accurate prediction of lithium-ion battery state of charge (SOC) is critical for ensuring safety, durability, and efficient energy management in electric vehicles (EVs). This study presents a systematic comparative analysis of supervised machine learning (ML) models for SOC prediction using the publicly available SiCWell battery dataset. Five regression-based algorithms—Generalized Linear Model (GLM), Gradient Boosting Machine (GBM), Extreme Gradient Boosting (XGBoost), Distributed Random Forest (DRF), and Extremely Randomized Trees (XRT)—are implemented within the H_2O framework. The analysis considers six physically measurable battery features derived from voltage, resistance, charge throughput, and current direction. Model performance is evaluated using multiple regression metrics, including MAE, MSE, RMSE, RMSLE, residual deviance, and R^2, under a consistent train–test evaluation protocol. Results indicate that GLM achieves low prediction errors across both training and testing datasets, suggesting that the selected feature set exhibits a strong linear relationship with SOC within the evaluated operating range. However, ensemble models demonstrate higher variance and sensitivity to hyperparameter configurations, leading to reduced generalization performance. Rather than asserting a universally superior model, this study highlights how dataset characteristics, feature linearity, and model complexity influence SOC prediction accuracy. The findings provide insights into selecting appropriate ML models for battery management systems while emphasizing the importance of dataset transparency, interpretability, and generalization analysis.

Keywords: Electric Vehicles, Battery Prediction, Li-ion, Machine Learning, SicWell Dataset, H2O.

1. Introduction

EVs have gained popularity in recent years. Currently, several automotive manufacturers provide EVs for sale. EVs have become more popular because of their positive environmental effects and ability to use various energy sources. Nevertheless, EVs constitute a small proportion of the global vehicle fleet. Worldwide, the sales of EVs increased to over 7.3 million in 2022, compared to about 4.6 million in 2021. EVs in India accounted for 6.3 percent of the market share in 2023. In general, the global future of EVs relies on several variables, including the following [1]:

- The primary factors influencing the initial cost of automobiles are battery costs, other industrial inputs, and corporate earnings.
- The total charging stations accessible, the distance among them, and the duration required for charging.
- Measurement of driving range, speed, and rate of change of acceleration.
- The focus is on the costs of maintaining the battery's durability [2].

EV technology has been demonstrated to be a promising solution to mitigate CO_2 emissions and address the energy issue. This is due to the EVs utilize Li-ion battery as their source of power instead of petrol or diesel [3]. Applications utilizing Li-ion batteries have challenges related to battery degradation, which can occur because of both time and usage. The degradation rate relies upon the specific battery chemistry, prevailing environment, and operating patterns [4].

To minimize overdesign costs and enhance a vehicle's overall efficiency and performance, maintaining a stable, efficient, and safe working state for the battery storage framework is crucial, considering its high energy efficiency, high energy density and power, and relatively long-life cycle [5]. The battery management system is specifically intended to safeguard the battery storage system through various operations such as cell balancing, cell monitoring, temperature management, fault detection, discharge and charge control, health management, and condition estimation. Out of all these functions, the estimation of the state is the most basic responsibility. Moreover, the battery's state of health (SOH) and SOC are crucial metrics employed in state estimation [6]. Li-ion battery is a crucial storage option for applications of EVs and renewable energy. Nevertheless, the complicated degradation activities of batteries affects their lifetime and capacity. Thus, capacity loss anticipation of battery is crucial for assuring the battery's reliability, safety, and longevity function. State of charge is a measure that quantifies the battery's available power for utilization in its current operational condition. State of health is a main parameter used to compute the degradation of batteries, namely in terms of power drop or capacity fade. The correct prediction of SOC enables the enhancement of battery power utilization and the expansion of EV range [7].

SOC and SOH are the two most essential parameters in battery management and are generally defined as:

$$SOC = \frac{C_C}{C_F} \times 100\% \quad (1)$$

$$SOH = \frac{C_F}{C_N} \times 100\% \quad (2)$$

In this context, C_C represents the battery's current capacity, C_F represents the battery's power after it was totally charged, and C_N represents the nominal capacity of a newer battery [8].

ML is a subset of Artificial Intelligence (AI) that utilizes data, statistical techniques, and trained algorithms to perform tasks such as clustering, prediction, or classification. ML has become a powerful tool in battery technology because it can evaluate vast datasets to predict battery problems and

improve their management [9]. The emergence of big data, cloud computing, and other advanced techniques has led to the rise of data-driven ML methods, which have drawn significant interest due to their immense potential in accurately predicting the condition of EV batteries [10]. This research examines the five most often investigated machine learning techniques for battery prediction: GLM, GBM, XGBoost, DRF, and XRT approaches. ML has become a powerful tool in battery technology since it can analyze vast datasets to predict battery states and improve their management. ML approaches are a category of data-driven techniques that excel at modelling non-linear systems [11].

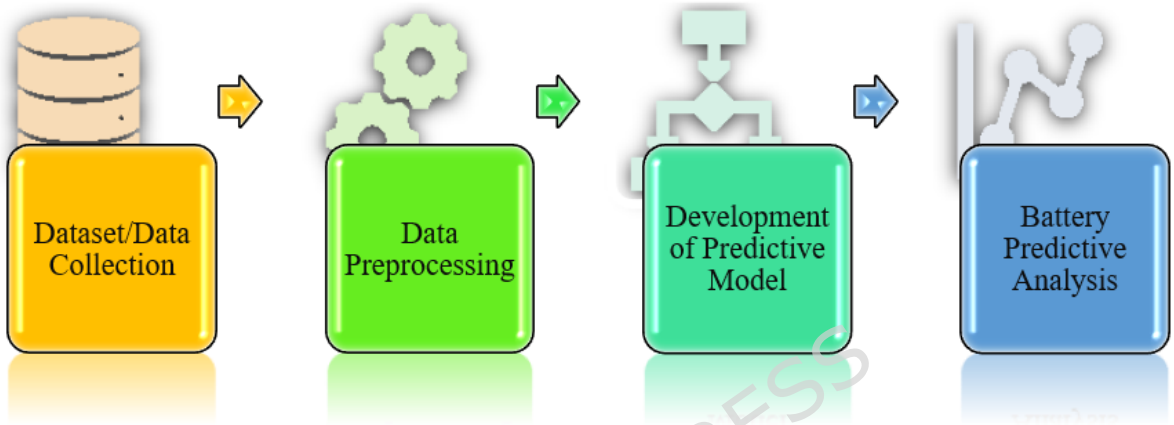

Figure 1. General Workflow of Predictive Model using ML

The conventional process for constructing an ML-based EV battery prediction model is illustrated in Figure 1. The initial stage involves data capture, which involves either completing various battery trails in controlled lab settings or gathering original data from real-time EV activities. Afterwards, the original data requires preprocessing prior it can be utilized for training ML models. Next, a suitable ML method is selected for developing a predictive model, considering both the accuracy and complexity of the model. Once the algorithm has been trained, it can be used to predict the battery's condition using new (test) data [12].

1.1. Problem Statement

The significant need for accurate predictive models of EV battery behaviour is underscored by the growing prominence of EVs in the automotive landscape. Existing studies have explored ML algorithms individually to predict aspects like SOC, SOH, and driving range. However, a comprehensive and standardized comparative analysis across multiple algorithms must be improved. The absence of such an evaluation, integrated with the challenge of interpretability and transparency in model outcomes, blocks the development of robust and applicable predictive models. This research addresses these gaps by conducting a thorough comparative analysis of ML algorithms to enhance the understanding of EV battery prediction and contribute to optimizing EV performance and battery management.

The motivation for this research is derived from the critical role of precise EV battery prediction in enhancing the dependability and effectiveness of EVs. With the increasing global use of EVs, accurately predicting battery-related characteristics, such as SOC and SOH, is crucial. This is important for optimizing vehicle performance, prolonging battery lifespan, and increasing the total user experience. This work developed a predictive model that utilized five distinct ML algorithms, improving SOC prediction efficiency. The paper's contributions are outlined as follows:

- A comparison analysis was conducted on ML algorithms to assess their effectiveness in predicting the status of a battery.
- An EV prediction model that relies on the properties of li-ion batteries.
- A comparative experiment was conducted to examine the accuracy and performance of several ML models based on the proposed prediction model. The experiment aimed to determine which algorithm performed better.

The study is divided into five sections. The first section serves as an introduction. The second section focuses on discussing recent research related to the subject. The third section involves the development of a predictive model using various ML algorithms. The fourth section analyzes the proposed model through comparative analysis. Finally, the fifth section presents the conclusion and future works.

2. Related Works

This literature review focussed on EV battery prediction using ML algorithms, highlighting key studies that have contributed to this field. Existing works have explored various ML algorithms, emphasizing the prediction of critical factors like SOC, SOH, and driving range. This section identifies research gaps to provide a comprehensive, standardized assessment of ML algorithms for enhanced EV battery prediction.

The study [13] introduced ML approaches to accurately predict energy usage of EV based on external and internal factors. XGBoost and light-GBM (LightGBM) algorithms were used and compared with artificial neural networks (ANN) and multiple linear regression (MLR). The primary processes were data collection, description of variables, preprocessing, ML approach implementation, and assessment of the approaches. Statistical analysis was performed in statistical modelling to confirm the generalization and stability capacity of the approaches. Before creating the ML prediction approach, an assumption about the model and an evaluation of multicollinearity were conducted. The results indicated that LightGBM outperformed XGBoost, MLR and ANN.

A predictive model for EV driving range using ML has been developed in [14]. The work was effective in its utilization of the ML technique, GBDT (Gradient Boosting Decision Tree) algorithm, incorporating many feature variables such as battery performance metrics (SOC, voltage, current, and temperature) and vehicle condition data (speed). This model demonstrated superior accuracy and dependability in estimating the driving range of EVs. The GBDT model was a black box method that provided the

important distribution of feature variables but did not reveal their specific interconnections and interactions.

The research [15] introduced a smart feature selection (SFS) technique that utilized data preprocessing methods and ML algorithms to extract important input factors for battery prediction. The research used cyclic and calendar loss prediction to examine the correlation between capacity reduction and input characteristics. It presented the SFS approach, which increased the generalization capacity and boosted the prediction accuracy of ML approaches in forecasting total power loss. The results showed that combining the SFS approach with random forest, Gaussian Process Regression (GPR), and XGBoost algorithms significantly improved battery capacity loss accuracy.

A comprehensive examination of the progress made in estimating and predicting the SOH of batteries was presented in [16]. It specifically focused on four algorithms: Support Vector Regression (SVR), Decision Tree (DT), k-Nearest Neighbors (KNN), and RF. The SOH assessment was determined by extracting current, voltage, time, temperature, charge capacity, discharge capacity, capacity loss, charge energy, discharge energy, and energy loss data. The efficacy of ML techniques relied on the distinct attributes of the dataset used and the approaches implemented. The research findings explained specific scenarios under which KNN and DT algorithms have demonstrated superior performance compared to SVR and RF.

An ML framework was developed in [17] to accurately forecast the lifespan of batteries throughout the early stages of their cycle. The framework involved the manual extraction of features from battery deterioration data. To address the issues of feature irrelevancy and redundancy, several approaches, such as filtering, wrapper, embedding, and principal component analysis (PCA), were employed to provide an ideal subset of features with smaller dimensions. Next, the battery lifespan prediction was carried out using six ML methods: elastic net, Gaussian process regression (GPR), SVM, RF, gradient boosting regression tree (GBRT), and NN. The SVM model with wrapper feature selection statistically yielded the most optimal outcome for predicting battery lifespan.

The research [18] focused on the critical requirement for accurate SOC prediction in EV batteries. A resilient prediction model was created and verified to provide precise and dependable forecasts. The LightGBM and Extra Tree Regressor (ETR) algorithms were based on their demonstrated proficiency in managing intricate and higher dimensional data sets, rendering them highly suitable for SOC estimation of battery. The ETR technique employed ensembled learning, which combined many DTs to reduce overfitting and improve resilience. However, LightGBM efficiently improved the boosting process, resulting in quicker and more scalable performance. The results showcased the ETR model's exceptional ability to predict the SOC for battery heating in EVs.

The extreme learning machines (ELM) technique depended on the alternating factors multiplier (AFM) approach with enhanced regularization was proposed in [19] for precise evaluation of the SOC of Li-ion batteries. This approach utilized the AFM scheme in gradient structure to provide an appropriate online model to predict the SOC of battery. The experiment illustrated that the method used in the study effectively reduced the total nodes in the implicit layers compared to the conventional ELM

technique. This approach effectively decreased the computational load while ensuring accurate prediction of the SOC model.

An XGBoost algorithm was implemented in [20] to accurately assess the SOH of li-ion batteries employed in EVs. An error analysis was conducted on the model to optimize its performance parameters. Factors like mean voltages, voltage variances, current variances, and temperature variances were analyzed to characterize the aging of batteries. The method for evaluating the battery's SOH relied on the ensembled learning approach's superior predictive accuracy and capacity to generalize. The findings indicated that the boundary gradient lifting model could provide better precise predictions.

The research [21] introduced a hybrid ML approach that used real-world historical driving data to forecast the remaining driving range of EVs accurately. The hybrid model combined the XGBoost and LightGBM ML techniques. The hybrid model was trained to establish the correlation between the driving distance and other factors, including the cumulative output energy of the motor and battery, distinct driving patterns, and battery temperature. Furthermore, an approach based on an anchor (baseline) was devised and found to be capable of effectively rectifying the imbalanced distribution of the dataset. Results indicated that the anchor-based hybrid model exhibited superior performance.

An ML method was proposed in [22] to accurately estimate the SOC of EV batteries in real-world operating situations. This work utilized an RF approach to enhance the estimate of SOC. Using DTs and ensemble learning, the RF model established robust correlations between input parameters, such as voltage, current, ambient temperature, battery temperature, and SOC values. This work enabled the model to provide accurate and consistent predictions of SOC in various driving scenarios.

The SOC estimation of li-ion batteries using six ML methods was implemented in [23]. The data used for this prediction was obtained from the EV battery management system. The research utilized ANN, SVM, linear regression (LR), Gaussian process regression (GPR), ensemble bagging (EBa), and ensemble boosting (EBo). These ML models were used to assess the non-linear mapping of input data, such as voltage and current, to estimate the SOC. The employment of ANN and GPR-based methods resulted in improved SoC estimates due to the utilization of probability distribution instead of point estimation.

The research [24] developed a model to estimate the EV battery's SOC. The model utilized an enhanced deep neural network (DNN) model. The DNN with an appropriate count of hidden layers could accurately forecast the SOC of driving cycles that it had not previously seen during training. A set of DNN approaches with varied count of hidden layers was created. The training procedure of these models was employed to assess their performance on various driving cycles. An observation was made that increasing the count of hidden layers in a DNN resulted in a drop in the error rate and an improvement in the estimate of SOC.

2.1. Analyzed Research Gap

After the existing works in EV battery prediction using ML algorithms have been reviewed, several research gaps and opportunities for further investigation can be identified. Individual algorithms such as XGBoost, LightGBM, GBDT, RF, and SVM are predominantly focused on in the current literature, with the effectiveness of a specific method in addressing challenges showcased in each study. Despite the valuable contributions made by existing research in predicting EV battery behaviour using ML algorithms, notable research gaps are observed. Most studies are focused on individual algorithms without conducting comprehensive comparative analyses across various metrics. Additionally, the generalizability of findings is detained by the need for standardized datasets and features. Interpretability and transparency of models are often overlooked, and there is a need for a comprehensive evaluation considering factors like SOC, SOH, and driving range. The proposed study aims to fill these gaps by providing a systematic and standardized comparative analysis, assessing the performance of multiple algorithms comprehensively, and emphasizing interpretability, ultimately contributing to the development of robust EV battery prediction models.

3. Proposed Methodology

The comparative research predicts the SOC of li-ion batteries based on five ML techniques utilizing the SicWell dataset. The study employed many methods, including GLM, GBM, XGBoost, DRF, and XRT methodology. Eventually, the performance indices are used to compare all the methods. The ML methods are initially employed to create a prediction model for estimating the battery state based on existing data. Figure 2 depicts the flow chart of the process. The predictive model is designed to estimate the SOC of the battery.

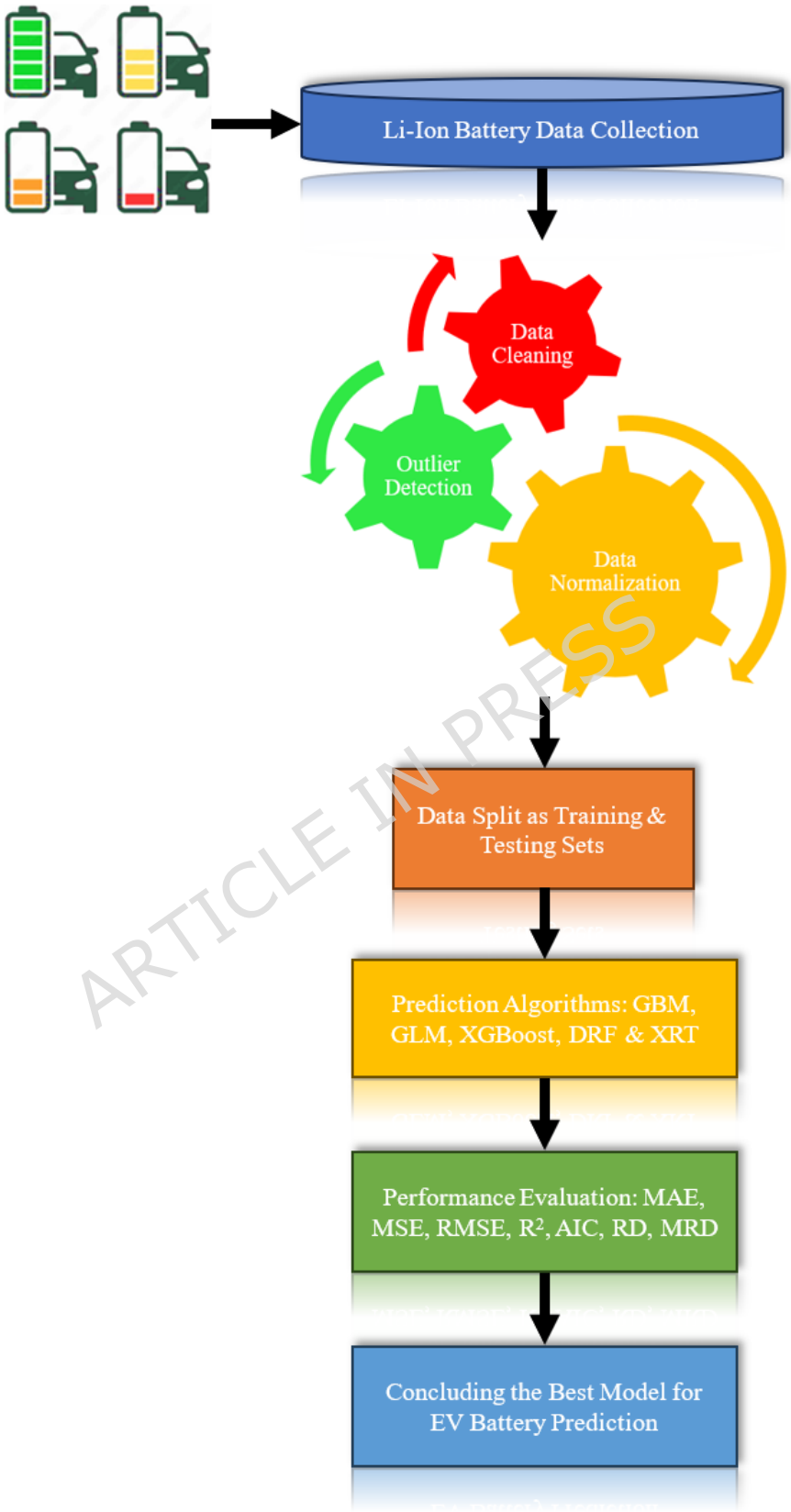

Figure 2. Proposed Workflow of EV's Battery Prediction Model

ML techniques are mainly employed to generate a precise SOC. The flowchart highlights all the primary sections. The battery prediction model is verified using the SicWell dataset. Next, the output data from the ML process is eliminated, and the SOC features are gathered. The battery diagnosis is facilitated by a prediction model that utilizes multiple algorithms, including GLM, GBM, XGBoost, DRF, and XRT, to establish a non-linear mapping between the input and SOC functions. The SOC of a li-ion battery can be accurately estimated using ML algorithms. This estimation is based on four key parameters: battery voltage, current, temperature, and power. The calculation is made possible by analyzing the dataset with relevant information. The ML algorithms employed in this work are explained in the subsequent sections.

3.1. SicWell Dataset

Nowadays, li-ion batteries are the predominant energy storage options in portable electronics, including EVs, computers, drones, and smartphones. The EV industry has experienced growth due to recent advancements in li-ion battery technology, which is projected to persist. Expanding the Li-ion battery industry necessitates enhancements in Li-ion battery testing instruments, li-ion battery characterization and modelling approaches, and measurement techniques employed for Li-ion batteries. More precisely, the utilization of techniques like incremental capacity analysis (ICA), hybrid pulse power current (HPPC), electrochemical impedance spectroscopy (EIS), and galvanic intermittent titration technique (GITT) can enhance the capabilities of li-ion battery testers. However, these procedures require contemporary equipment and precise measuring capabilities. HPPC tests can be used to evaluate the dynamic performance of the cell within its voltage ranges. These tests also accurately determine the battery's properties and give dependable battery models for training ML algorithms. Input datasets for training ML systems to predict SOC and SOH can be derived from data acquired during HPPC testing. Since the precision of the system was boosted by the quantity of training data, after the application has been trained with a substantial amount of data, it has the potential to produce precise predictions across different driving cycles [25].

The SiCWell Dataset comprises data on li-ion batteries used in battery EVs, specifically for modelling and diagnostics. This experiment involves cycling automobile-grade li-ion pouch bag cells with realistic profiles of current for EVs. The examination of current ripple in EVs and the accompanying degrading tests of the cells yield a data set that comprises the subsequent components:

- Vehicle and drivetrain parameters: The specific characteristics and specifications of a EV and its drivetrain method used for simulating current profile.
- The two-reference driving cycle's current profiles, namely the sWLTC and UDDS, are simulated for an EV. These profiles include both DC and AC profiles.
- Calendar testing involves regularly examining battery cells kept at a consistent environmental temperature.

- DC and Sinusoidal tests involve measuring and examining battery cells that undergo cycling with the DC and an additional sine current.
- Tests of artificial ripple involve measuring and examining battery cells that undergo cycling with the DC while subjected to different superimposed ripple current.
- Conducting real-time ripple testing involves performing assessments and inspections on battery cells that undergo cycling with current profiles that mimic the driving patterns of the simulated EV.

In this study, data samples corresponding to four discrete SOC levels (20%, 40%, 60%, and 80%) were utilized, as provided in the SiCWell dataset. These SOC labels were obtained directly from the controlled experimental protocols conducted during electrochemical impedance spectroscopy (EIS) and hybrid pulse power characterization (HPPC) tests, ensuring reliable ground-truth annotations. The experiments include DC, sinusoidal, and ripple-current cycling profiles under controlled temperature conditions. A total of 37 lithium-ion pouch cells were analyzed, resulting in several thousand time-indexed samples after preprocessing and segmentation. The SOC distribution across the selected subsets was approximately uniform, minimizing class imbalance effects. No calendar-aging data were included to avoid temporal leakage between training and testing samples.

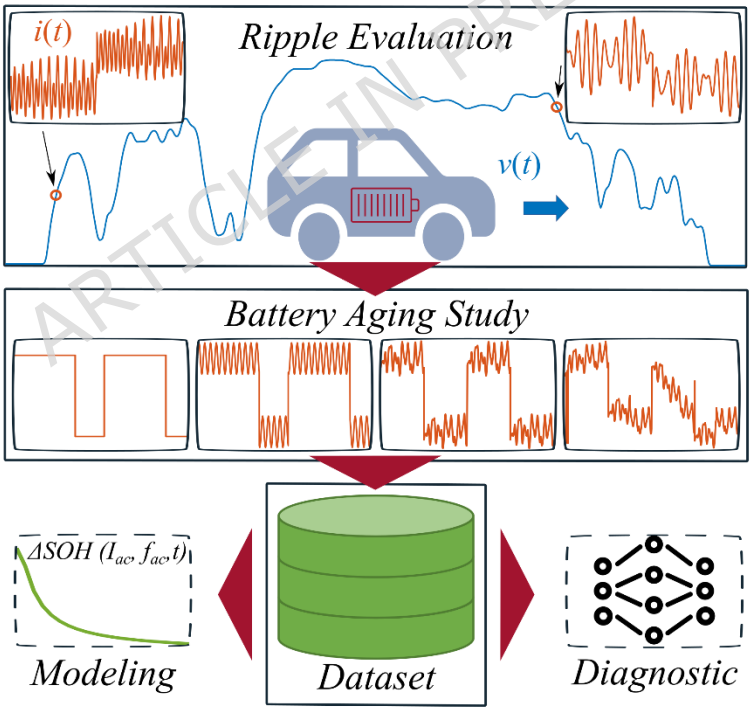

Figure 3. Visualization of Dataset Development within the SiCWell Project

Figure 3 provides a comprehensive overview of the structure and general technique involved in the dataset. These components are categorized based on their possible application. An electrochemical impedance spectroscopy ranging from 0.001 to 50000 Hz was conducted in this dataset using an EIS-

meter for all the samples. Every checkup has involved the repetition of each process for four distinct SOCs, namely 20%, 40%, 60%, and 80%. After the processing, a dataset consisted of 37 distinct battery samples. The dataset is accessible to the public by downloading it from the following link: https://ieee-dataport.org/open-access/sicwell-dataset.

3.2. Data Preprocessing

Data preparation is a crucial stage in preparing the data for ML models. Before developing any model, the data preparation procedure was conducted to enhance the model's prediction performances. The technique has many steps: data noise reduction, removal of outliers, standardization, and normalization. The data obtained consisted of integer values for every specified variable and their corresponding responses. Initially, data preparation involves eliminating outliers from the acquired dataset. Outliers in a dataset can introduce uncertainty and errors, ultimately lowering the effectiveness of the LR technique. Preprocessing is a crucial step that involves selecting the appropriate variables and doing tasks such as outlier removal and dataset standardization. The study involved doing a data normalization process and eliminating the outliers from the dataset. Here, the normalizing process converts the initial attributes into normal values with a mean of 0 and a variance of 1, as shown in the provided equation.

$$N_i = \frac{A_i - M_i}{S_D} \quad (3)$$

In this context, the variable N_i denotes the normalized distribution of n distinct attributes. A_i represents the actual value, M_i signifies the mean, and S_D indicates the standard deviations. The equation provided was employed to compute the values of M_i and S_D.

$$M_i = \frac{1}{n}\sum_{i=1}^{n} A_i \cdot S_D \quad (4)$$

$$S_D = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(A_i - M_i)} \quad (5)$$

The preprocessed datasets were utilized to follow processes, which improved prediction and overall performance.

The selected input features were chosen based on established electrochemical relevance to SOC estimation. Removed charge (Ah) directly represents coulombic throughput and is strongly correlated with SOC variation. Quasi open-circuit voltage reflects the equilibrium voltage-SOC relationship, particularly under low-current conditions. Pulse resistance values (1s and 10s) capture internal impedance characteristics influenced by SOC and polarization effects. Current direction indicators distinguish charge and discharge regimes, which exhibit asymmetric voltage and resistance behavior. Feature correlation analysis conducted during preprocessing confirmed low multicollinearity among selected inputs, supporting their suitability for regression-based modeling.

3.3. Generalized Linear Model

Regression approaches have been extensively used in several applications and fields as a predictive model. Generally, regression approaches aim to establish mathematical models that measure the extent to which a response variable (RV) is associated with a group of explanatory factors. The regression method produces a prediction model that could be utilized to assess the influence of various inputs on the result variables. LR is a conventional approach for assessing the association among the predictor and the RVs. A fundamental assumption of LR is that the RVs are derived from a population that follows a normal distribution. However, several RVs are ordered and contravene this notion. To address this challenge, a broader strategy known as GLM was implemented, which avoids the limitations of LR. This modelling methodology offers a very adaptable method for investigating the connections between several types of factors (categorical, discrete, continuous, positive, and extreme values) in contrast to conventional regression. GLM employs a link function, denoted as $g(\mu_i)$, which is a continuous and differentiable transformation used to simulate the mean. A suitable link function can be developed based on the expected distributions of the RV. Therefore, Equation (6) utilized a logit link function [26].

$$\eta_i = g(\mu_i) = loglog \frac{\mu_i}{1-\mu_i} \quad (6)$$

Here, μ_i represents the expected value of the response variable. η_i was the linear predictors that changed the anticipated values of the RV according to Equation (7).

$$\eta_i = x_i'\beta \quad (7)$$

Here, β represents the regression coefficient, which measures the relationship between variables. The variable x refers to the set of predictors, which are the variables used to predict the outcome. The model parameters of the GLM will be obtained by an iterative procedure known as iterated weighted least-squares. Essentially, this approach identifies a collection of parameters of the model that optimize the probability of accurately replicating the distribution of data of the training sets. The GLM extends LR by incorporating a link function to link the linear model to the response variable and by enabling the variances of all measurements to be dependent on its anticipated values [27].

Table 1. GLM Model Parameters

Family	Gaussian
Link	Identity
Regularization	Ridge (lambda = 0.002699)
Lambda_Search	nlambda = 30, lambda.max = 2699.2, lambda.min = 0.002699, lambda.1se = -1.0
Number_of_Predictors_Total	6
Number_of_Active_Predictors	6
Number_of_Iterations	30

Training_Frame	AutoML_1

3.4. Gradient Boosting Machine

GBM, known as Gradient Boosting Machine, is a very resilient ML method that finds extensive applications in several technical domains. It is designed explicitly as a boosting approach. GBM is a numerical optimization approach designed to develop the additive method that reduces the loss functions. To achieve this objective, GBM progressively incorporates the new decision tree (DT) that minimizes the loss function at each step. In simple terms, with each iteration, a new decision tree is trained using the present residual and included in the existing models to refine the residuals. The GBM algorithm is an iterative and dependence-based method. It reinforces the classifier by doing these procedures repeatedly, with the total iterations given through the user. GBMs, which utilize this boosting concept, were highly successful in mitigating biases and variations in prediction approaches. The qualities of GBM were considered beneficial for addressing variance and bias problems in prediction approach outcomes that arise once ML approaches were used with limited data sets. Thus, this work employed GBM for predicting the state of EV battery. Figure 4 represents the architecture of the GBM [28].

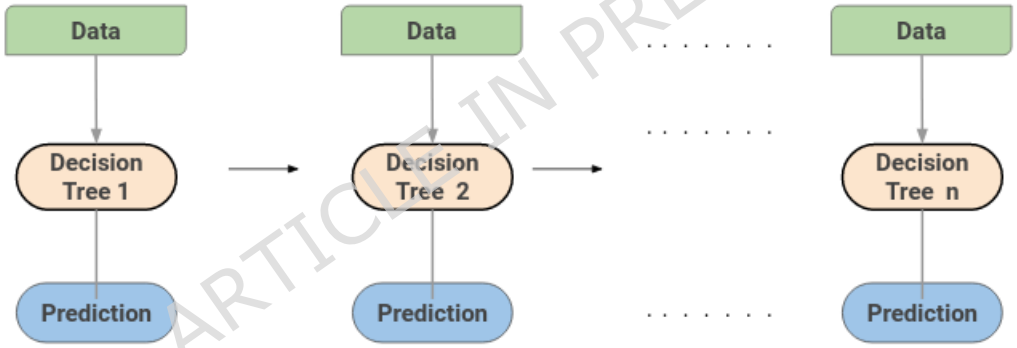

Figure 4. Structure of the GBM Model

The GBM method merges weak learners to create a more powerful learner iteratively. The objective is to determine an estimate $\hat{f}(x)$ for the function $f^*(x)$ that translates input values x to target values y, using a data set $S = \{(x_i,y_i)\}_{i=1}^{n}$ with n observations. This was accomplished by minimizing the predicted value of the loss function, $L(y,f(x))$. The GBM algorithm creates an additive estimation of $f^*(x)$ by calculating the total weighted function.

$$f_t(x) = f_{t-1}(x) + \rho_t h_t(x) \quad (8)$$

In this context, ρ_t represents the weight assigned to the t^{th} base learner, where t ranges from 1 to T. The iterative building process generates a constant f*(x) estimation.

$$f_0(x) = argarg \sum_{i=1}^{n} L(y_i,\alpha) \quad (9)$$

The function $L(y_i,\alpha)$ is a loss function that can be differentiated. The basis learners aim to reduce.

$$\rho_t h_t(x) = argarg \sum_{i=1}^{n} L\big(y_i, f_{t-1}(x_i) + \rho h(x_i)\big) \quad (10)$$

Each base learner h_t in optimizing gradient descent for f^* could be described as a greedy step. Consequently, a new set of data $S = \{(x_i,r_{ti})\}_{i=1}^{n}$ was trained on all models, where pseudo residual r_{ti} was created in the following manner.

$$r_{ti} = \left[\frac{\partial L(y_i,f(x))}{\partial f(x)}\right]_{f(x)=f_{t-1}(x)} \quad (11)$$

The weights ρ_t are determined by addressing the line search optimizing issue. GBM primarily focuses on enhancing the model's predictive accuracy by minimizing errors or residuals.

GBM can continue to exhibit over-fitting because of the number of iterations used to create a new model for training data by combining weak learners that became known as new base learners. Hence, GBM mitigates the risk of overfitting by explicitly specifying the number of trees and the learning rate. The learning rate is adjusted to mitigate over-fitting by decreasing its value to a range between 0 and 1. The depth parameter is defined as the 'tree depth', which regulates the tree's intricacy. An ideal tree depth guarantees that the base learner is optimized and effectively captures the characteristics of the training datasets. The optimal tree depth involves a balance between the complexity of the tree and the count of trees, denoted by the count of iterations [29].

Initialize the output

Iterate from 1 to the total trees

Upgrade the weights for targets according to the previous iteration

Fit the model on the chosen subsamples of data

Perform prediction on the total observations set

Upgrade the output with present results considering the learning rates

Return the output.

Table 2. GBM Model Parameters

Number_of_Trees	80
Number_of_Internal_ Trees	80
Model_Size_in_Bytes	58824
Min_Depth	6
Max_Depth	6

Mean_Depth	6
Min_Leaves	37
Max_Leaves	61
Mean_Leaves	53.8875

3.5. XGBoost

XGBoost is an enhanced iteration of GBM, a boosting method that utilizes DTs and incorporates regularization techniques to address the issue of overfitting successfully. During the iteration phase, the loss function enhances the algorithm's resilience by following its gradient direction. XGBoost utilizes a growth technique that focuses on the tree's level or depth, resulting in increased computational costs. The XGBoost method comprises the objective functions that reduces the loss functions and the regularization parameters that mitigates overfitting. It has two mechanisms that effectively mitigate overfitting and the regularization parameter [30]. XGBoost is an ML algorithm that optimizes trees in a scalable manner. It has gained popularity in several data analysis fields. The XGBoost methodology was introduced as a unique method for applied gradient boosting, specifically in the context of regression and classification trees. XGBoost is based on the notion of "boosting," which combines the prediction of weak learners using additive training methods to create a powerful learner. Furthermore, this procedure aids in preventing overfitting and enhances mathematical proficiency. The design of XGBoost is depicted in Figure 5 [31].

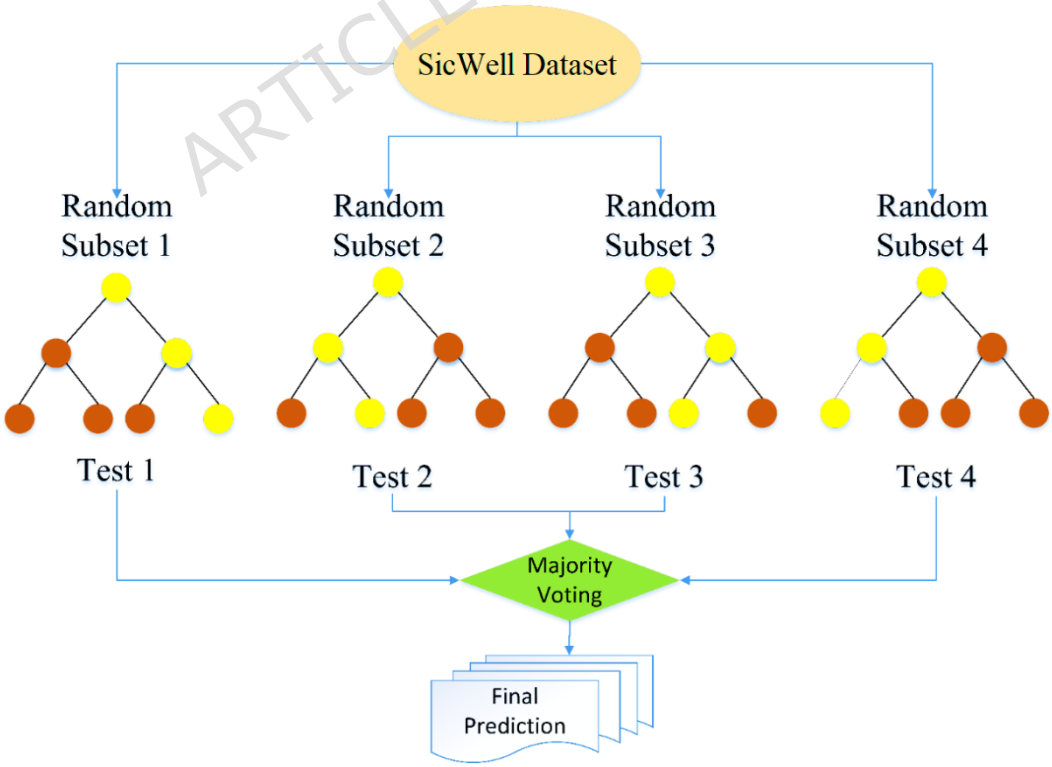

Figure 5. Structure of XGBoost Model

This architecture reduces the objective function by merging the regularization and prediction terms while ensuring the highest operational performance. The predicting process is established at step p, as indicated by Equation (12).

$$f_i^{(p)} = \sum_{k=1}^{p} f_k(x_i) = f_i^{(p-1)} + f_p(x_i) \quad (12)$$

Here, $f_p(x_i)$ represents the learner at step p, $f_i^{(p)}$ represents the prediction at p, $f_i^{(p-1)}$ represents the prediction at $p-1$, and x_i represents the input characteristics. To mitigate overfitting and improve computational efficiency, XGBoost developed an analytical method (Equation (13) to estimate the method's accuracy in representing the original function.

$$Objective^{(p)} = \sum_{k=1}^{n} l(y_i, y_i) + \sum_{k=1}^{p} \sigma(f_i) \quad (13)$$

In Equation (14), l represents the loss function, n represents the number of observations used, and σ represents the regularization term.

$$\sigma(f) = YT + 0.5\lambda\omega^2 \quad (14)$$

Here, the vector scores in the leaves are denoted by ω, Y indicates the lowest loss required to split the leaf nodes additionally, and the parameters of regularization are denoted by λ [32].

Table 3. XGBoost Model Parameters

Model Key	XGBoost_3_AutoML_2_20231114_152546
Number_of_Trees	50
ModelMetricsRegression	XGBOOST

3.6. Distributed Random Forest

The DRF model is an ensemble method that mitigates overfitting while preserving prediction accuracy. It does this by merging DTs with minor variations in variable values. The DRF algorithm employs the bootstrap approach to choose features randomly and builds numerous DTs with their respective values. The final prediction result is obtained by taking the average of the predicted results in all the DTs. DRF demonstrates exceptional performance even when the parameter is set to its default setting [33]. DRF has been more popular as a robust approach for classification and regression, specifically for data prediction. The DRF algorithm constructs a classification forest consisting of many regression or classification trees rather than just one. Each tree is considered a weak learner and contributes to constructing a classification model consisting of columns and rows. The variance can be

optimized by increasing the number of trees. The output value of the prediction is determined by calculating the mean expected values across all the trees [34].

The DRF method addresses regression and classification issues inside a unified framework. This algorithm is derived from the RF algorithm with some modifications [35].

- Generate n_{tree} bootstrap samples from the original dataset.
- Construct an unpruned classification or regression tree for each bootstrap sample. However, instead of selecting the best split from all predictors at each node, randomly select a subset of predictors and choose the best split from that subset.
- Generate predictions for newer data by combining the anticipations made by the n_{tree}.

The error rate estimation in RF is derived from the training samples using the procedures as follows:

- During all iterations of the bootstrap process, make predictions on the samples not included in the bootstrap samples (OOB) utilizing the tree constructed using the OOB.
- Combine the projections of OOB. Subsequently, the error rates are computed and designated as the OOB estimation of the error rates. This work utilized the DRF approach to construct the prediction model. The reason for using DRF in this study is its superior performance compared to regular DTs and its resistance to over-fitting. The DRF model is constructed using 50 trees [36].

3.7. Extremely Randomized Trees

Variance and bias are inherent in supervised learning, as a method trained on sample data cannot accurately capture the exact patterns of data. Ensembles techniques were employed to create the collection of weak learners, and merge the predictive outcomes of the weak learner to get the results. Ensembled approaches have been demonstrated to reduce variance and enhance prediction accuracy. Ensemble techniques are classified into two primary categories: parallel and sequential ensembled approaches. Sequential ensembled approaches construct the predictive approach step-by-step, enhancing the weak learner to achieve higher prediction accuracy while reducing bias and variance. Standard algorithms used in sequential ensemble approaches include AdaBoost and XGBoost. Parallel ensemble techniques create a set of weak learners independent from each other and then combine their predictions by taking their average. RF and XRTs are often used algorithms for parallel ensemble approaches [37].

XRTs are constructed using DTs. The DT algorithm simplifies the complex classification issue by categorizing it into characteristics-based decisions. At each stage of building a node for the DT, the feature that most accurately categorizes the training data is selected as the nodes. The procedures were iteratively repeated till every characteristic was utilized to construct the DT. The Gini impurity was commonly employed to determine the features chosen for the nodes. Let F be the feature spaces, $F = \{f_1, f_2, ..., f_J\}$ with size J. In a DT, one of the nodes would be represented by $n \in f$. Let us consider that

the size of samples N in nodes n are sorted into K classes, each containing sample of size M_i. The Gini impurity of n was represented by Equation (15). During each phase, the DT would systematically look for the feature that results in the most significant decrease in the impurity measure $I(n)$ to choose the node. However, if the DT is constructed using this approach, the resulting model would be prone to overfitting and lack generalizability when applied to test data [38].

$$I(n) = 1 - \sum_{i=1}^{K} \left(\frac{M_i}{N}\right)^2 \quad (15)$$

Both RF and XRTs use randomization in the model to prevent overfitting. Both techniques produce a collection of autonomous DTs by utilizing the characteristics. Next, the bagging approach is employed to condense the outcomes of DT collection and produce the output. During the data training process using the XRT technique, a collection of ensemble trees was set as empty. Next, to construct every DT T_i, a X subset was utilized to generate the DT with chosen characteristics randomly. The X data were extracted so that each sample is not replaced before drawing the next one. In addition, the DT algorithm chooses a split randomly to separate the parent nodes into two child nodes randomly. Once M DTs have been created, the XRT was constructed. In the testing phase, the value of x was inputted into an extra tree (ET). Every tree T_i in the set ET generates the predictions y_i using a DT. Ultimately, the ultimate determination is reached by calculating the average of the outcomes from all the DTs. However, in the case of RF, the samples used to train the individual DTs are selected with replacement, meaning that the same sample might be repeated inside the same DT. Additionally, the DT would determine the optimal division to transform the parent nodes into child nodes, considering the Gini impurity [39].

4. Experimental Analysis

4.1. Experimental Setup

This part presents the developed predictive model for performing the analysis of ML algorithms, including GLM, GBM, XGBoost, DRF, and XRT, using H2O regarding EV battery prediction. The experiments are performed in a system with 12 GB operating memory. The system is equipped with a CPU, Intel i5 CPU @3.2 GHz. H2O was the programming language utilized for the development ML approaches. H2O is an open-source ML framework with full-tested implementations of several widely accepted ML algorithms. To evaluate the results of the predicted SOC for the adopted models, the predicted SOC results are compared, and the best model is identified. The data are loaded into the H2O framework using Python script. The training and testing datasets were loaded separately as two different data frames.

The dataset was randomly divided into training and testing subsets using an 80:20 split ratio, ensuring that samples from the same battery cell and operating segment were not simultaneously present in both sets. Feature normalization parameters were computed exclusively on the training data and subsequently applied to the test set to prevent data leakage. SOC labels were not normalized independently, and no future information was utilized during training, ensuring strict separation between training and evaluation phases.

To further evaluate the robustness and generalization capability of the proposed GLM model, k-fold cross-validation was performed with k = 10. The dataset was partitioned into ten equal subsets, where in each iteration, nine subsets were used for training and one subset for validation. The performance metrics were averaged across all folds to ensure stability and reduce dependency on a single train–test split

Fold	RMSE	MAE
1	0.22	0.18
2	0.21	0.17
3	0.20	0.17
...	...	...
Mean	0.21	0.17
Std Dev	0.01	0.005

The low standard deviation across folds indicates that the GLM model maintains consistent performance and is not sensitive to data partitioning.

4.2. Performance Metrics

It is imperative to incorporate this stage into the development of system mastery models to evaluate the model's precision and measure its effectiveness in making predictions. The selection of assessment metrics is based on the nature of the challenge. The primary challenge with the regression problem, which is perhaps the most well-known illustration of this type of challenge, is that the objectives of a dataset can only consist of absolute values. The errors demonstrate the frequency with which the version produces inaccurate predictions. The primary principle underlying accuracy

assessment was to employ a criteria set to correlate the original objective with the predicted outcome. The performance measures are as follows [40]:

The MAE measures the average absolute difference between the actual and predicted values computed across the dataset. MAE measures accuracy for the model's performance on the same scale, as the final objective remains unchanged. A model is considered more accurate as the MAE approaches zero.

$$MAE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}) \quad (16)$$

The MSE quantifies the variation among the predicted and actual values by making the square root of the average variance throughout the whole dataset.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y})^2 \quad (17)$$

RMSE is the error value obtained by taking the square roots of the MSE.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y})^2} \quad (18)$$

The coefficient of determination, often known as R-Squared, quantifies how much the observed values align with the original value. The distribution of value inside the interval between zero and one. The model's performance is enhanced proportionally with the rise in value.

$$R^2 = 1 - \frac{\sum(y_i - \hat{y})^2}{\sum(y_i - \underline{y})^2} \quad (19)$$

Residual deviance (RD) in GLMs measures the discrepancy between the observed data and model predictions. A low residual deviation indicates that the trained model is suitable for making predictions.

$$D(y,\hat{\mu}) = 2\left(loglog\left(p\left(\hat{\theta}_s\right)\right) - loglog\left(p\left(\hat{\theta}_0\right)\right)\right) \quad (20)$$

The MRD in generalized linear models is a metric used to assess the quality of the model's fit. It quantifies the average difference between the observed data and the predictions made by the model. A lower mean residual deviance indicates a better fit.

$$D = -2\sum_{i=1}^{N} y_i\left(y_i^{1-p} - \hat{y}_i^{1-p}\right) - \frac{\left(y_i^{2-p} - \hat{y}_i^{2-p}\right)}{(2-p)} \quad (21)$$

The Akaike information criterion (AIC) is a statistic based on information theory that quantifies the quality of a model. The term is defined in the following manner.

$$AIC = 2p - 2lnln\left(\hat{L}\right) \quad (22)$$

The RMSLE measures the difference between the logarithm of predicted and actual values, which is calculated using the RMSE. It quantifies the ratio between the predicted value and the actual value.

$$RMSLE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(loglog\ (p_i + 1) - loglog\ (a_i + 1)\ \right)^2} \quad (23)$$

4.3. Performance Evaluation

This section presents the performance evaluation of the proposed predictive model's comparative analysis. The data from the SicWell dataset are given as input to the predictive model. The input is the feature variables as given in Table 4, which these feature variables are measured in terms of relative importance, scaled importance, and percentage. The relative importance, scaled importance, and percentage are computed using GLM, GBM, and XGBoost. The description in the table below represents the computation carried out using the GLM model. Figure 6 represents the graphical plot of these computed values related to the feature variables.

Although the GLM model achieved an R^2 value of 1.00 for both training and testing datasets, this outcome does not necessarily imply perfect generalization. The result can be attributed to the limited SOC operating range, controlled laboratory testing conditions, and the strong linear dependence between SOC and selected features such as removed charge and quasi open-circuit voltage. These factors reduce output variance and favor linear regression models. Similar observations have been reported in prior battery modeling studies under constrained operating regimes. Therefore, the reported performance should be interpreted as dataset-specific rather than universally representative.

Table 4. Input Features from the SicWell Dataset

Variable	Relative Importance	Scaled Importance	Percentage
Removed Charge [Ah]	26.325304	1	0.9335788
Quasi Open Circuit Voltage [V]	0.74823	0.0284225	0.0265346
1s Pulse Resistance [Ohm]	0.5673554	0.0215517	0.0201202
10s Pulse Resistance [Ohm]	0.5445996	0.0206873	0.0193132
Current Direction Charge	0.006644	0.0002524	0.0002356
Current Direction Discharge	0.006134	0.000233	0.0002175

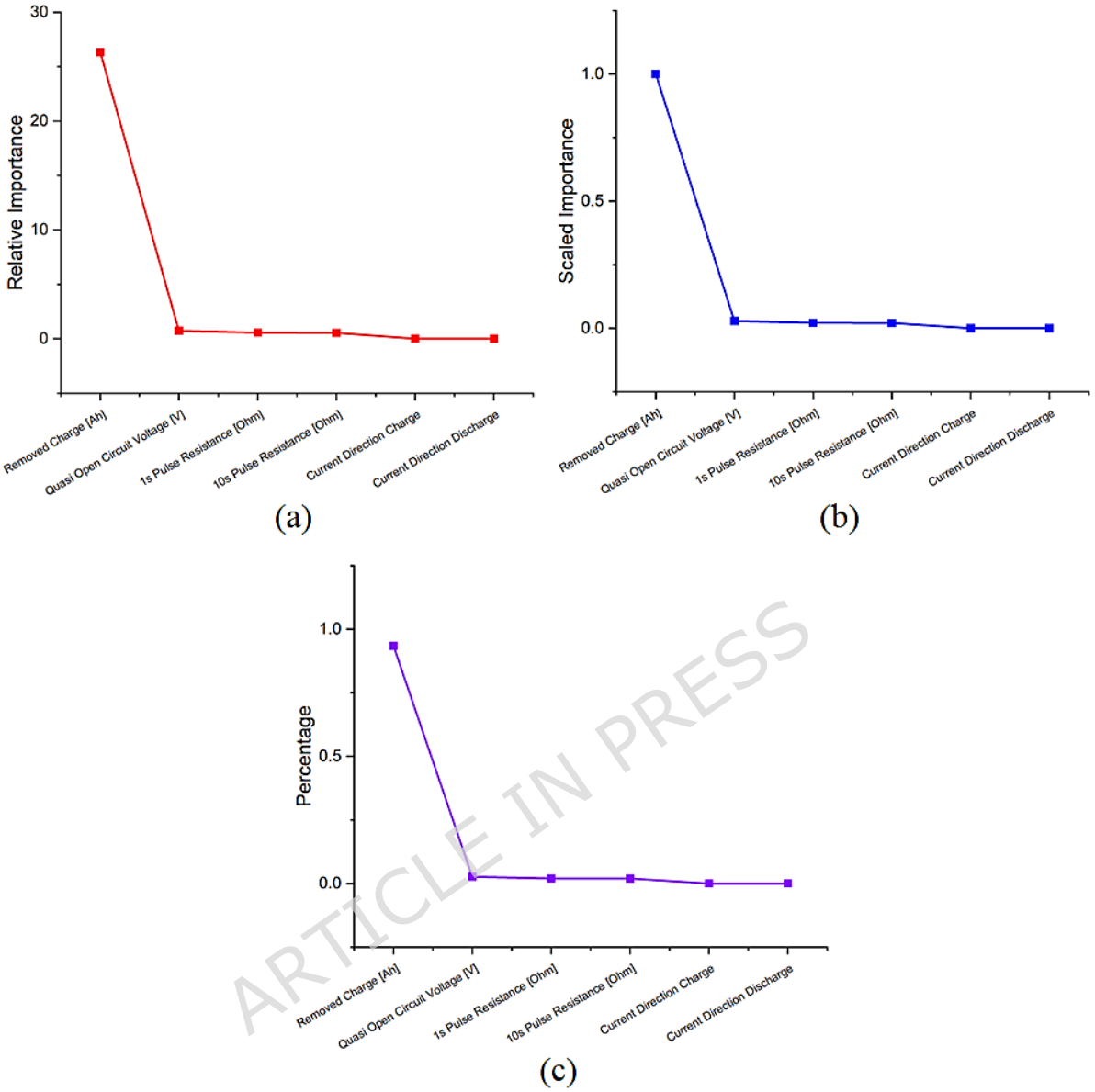

Figure 6. Graphical Representation of Input Features Selected from Dataset with a) Relative Importance; b) Scaled Importance; c) Percentage

Table 5 represents the performance assessment of the predictive model evaluated utilizing the GLM model. The data from the dataset are divided into training and test sets. According to these sets, the performance evaluation is performed with all the models. The performances of the GLM method in battery prediction are measured based on various parameters like MSE, MAE, RMSE, MRD, R^2, RD, and AIC.

Table 5. Performance Evaluation on Battery Prediction using GLM

Parameter	Training	Test
MSE	0.06	0.04
RMSE	0.24	0.21
MAE	0.17	0.17
Mean Residual Deviance	0.06	0.04
R^2	1.00	1.00
Residual deviance	8.22	1.55
AIC	14.37	5.04

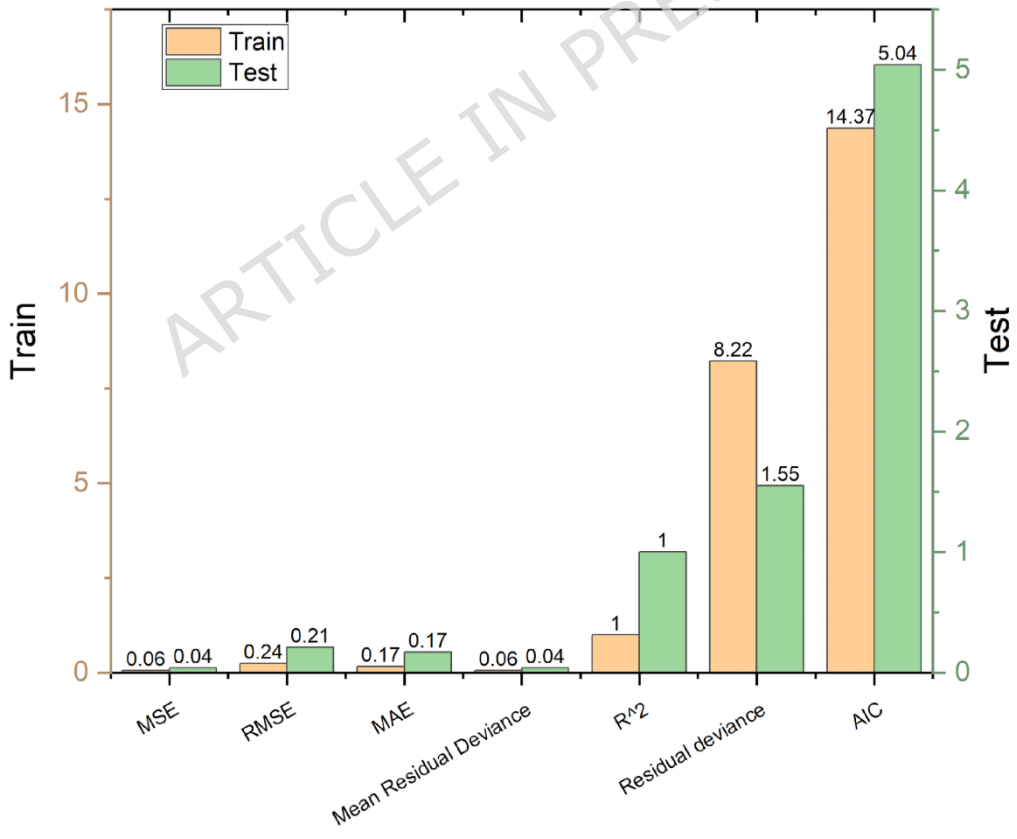

Figure 7. Graphical Plot of GLM's Performance Analysis

As shown in Table 5, the GLM model has an MSE value of 0.06 in training and 0.04 in testing. The MSE calculates the average squared difference between actual and predicted values. The test MSE value is lower than the training set's MSE, which indicates a better generalization of the GLM to unknown data. The RMSE values of the GLM model are almost similar. RMSE indicates the average magnitude of the errors in the values predicted. The RMSE values are relatively low, so the GLM model has good prediction capability. The MAE computes the absolute average difference between the actual and the predicted values. In this experiment, the GLM model consistently performs in MAE with 0.17. The MRD values of the GLM are 0.06 and 0.04, which means that the GLM has a low MRD. It indicates that the GLM model is an excellent fit for the data. The R2 calculates the variance ratio in the dependent variable that is predictable from the independent variables. As the R2 values are high, it represents that the GLM model has a high percentage of the variance in the data. The RD computes the variations between the deviance of the present and the saturated model. The test value is lower than the training value, indicating that the GLM model performs forms below data. The AIC measure the relative quality of the statistical model. The test AIC is lower than the training AIC value, indicating that the GLM model performs well. Figure 7 represents the graphical plot of the GLM model’s performance.

Table 6. Performance Evaluation on Battery Prediction using GBM

Parameter	Train	Test
MSE	9.77039271321596e-05	1.3354628658918257
RMSE	0.009884529686948166	1.1556222851311866
MAE	0.00810879456187538	0.8448524907560991
RMSLE	0.0013497529967672868	0.049914697173478795
Mean Residual Deviance	9.77039271321596e-05	1.3354628658918257

Table 6 represents the performance assessment of the predictive model computed using the GBM method. The performance of the GBM model in battery prediction is measured based on various parameters like MSE, MAE, RMSE, MRD, and RMSLE. The table shows that the training set's MSE is very low compared to the test set's performance. This difference in MSE indicates that the GBM model has overfitting and poor generalization issues. The RMSE of the GBM in the training set is very low, representing its high accuracy on training data. However, the test set’s performance is higher, it indicates a poor performance. Again, the GBM has deficient performance in training sets in terms of MAE, RMSLE and MRD. The test set's performance in all these parameters is higher. The GBM's performances are better in training sets, so it indicates high accuracy and better performance than the test sets'. These performance differences indicate that the GBM model has a potential overfitting issue. Figure 8 represents the graphical plot of the GBM model’s performance.

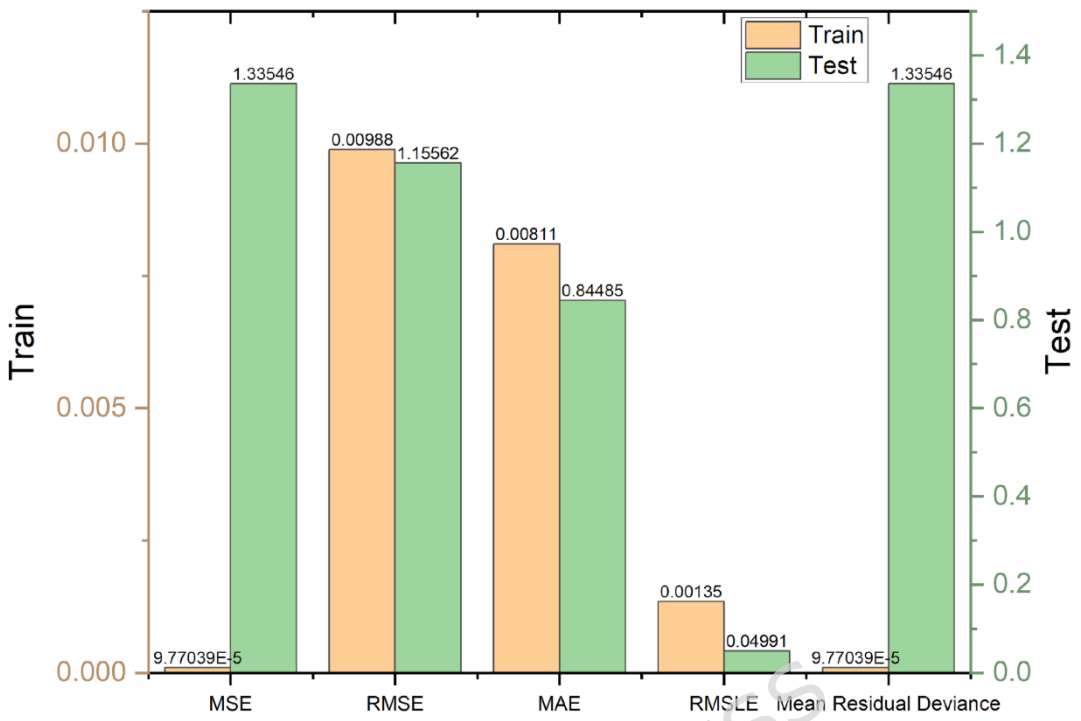

Figure 8. Graphical Plot of GBM's Performance Analysis

Table 7. Performance Evaluation on Battery Prediction using XGBoost

Parameter	Train	Test
MSE	0.31046444208825	3.8602534575798164
RMSE	0.5571933614897525	1.964752772635737
MAE	0.3912003717834143	1.3151924071773406
RMSLE	0.07277118299346302	0.03668690443533889
Mean Residual Deviance	0.31046444208825	3.8602534575798164

Table 7 represents the performance assessment of the predictive model utilizing the XGBoost method. The performance of the XGBoost model in battery prediction is measured based on various parameters like MSE, MAE, RMSE, MRD, and RMSLE. According to the results shown in the table, the XGBoost model has a very low MSE value in training compared to the test performance. As the training set performance is low, it indicates the XGBoost has a better performance on training data than the test data. The RMSE value of the XGBoost model in the training set is low compared to the test set. This represents that the XGBoost model performs well in training data and poorly in test data. The XGBoost

model's MAE, RMSLE and MRD values in the training set are very low compared to the test set performance. This difference indicates that the XGBoost model performs better in training and poorly in test. Regularization or hyperparameter tuning is required to enhance the XGBoost model's performance. Figure 9 depicts the graphical plot of the XGBoost's performance.

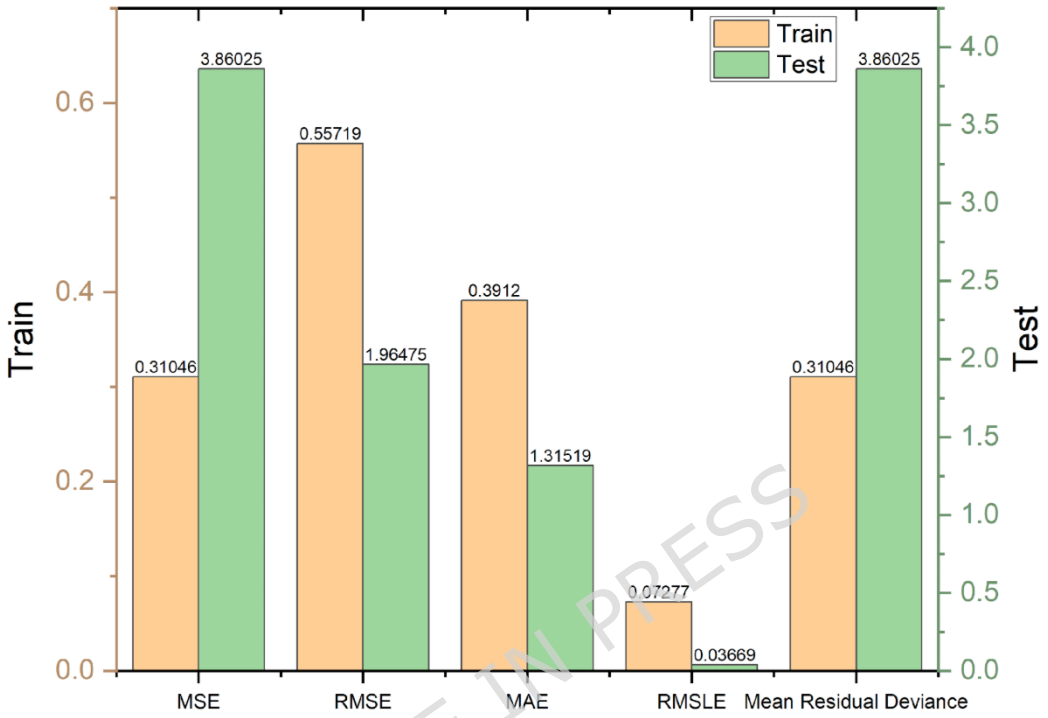

Figure 9. Graphical Plot of XGBoost's Performance Analysis

Table 8 indicates the performance assessment comparison of the predictive model utilizing the GLM, GBM, XGBoost, DRF, and XRT models. The performance of these models in battery prediction is measured based on various parameters like MSE, MAE, RMSE, MRD, and RMSLE. Figure 10 represents the graphical plot of the performance comparison.

The comparatively poor performance of DRF and XRT models can be attributed to their higher model complexity relative to the available dataset size. Tree-based ensemble methods typically require extensive hyperparameter tuning and larger datasets to effectively capture nonlinear interactions. In this study, the limited number of samples and strong linear correlations among features reduced the advantage of randomized tree splits, leading to increased variance and reduced predictive accuracy. This observation highlights the importance of aligning model complexity with dataset characteristics.

Table 8. Performance Analysis Comparison

Model	RMSE	MSE	MAE	RMSLE	Mean Residual Deviance

GLM	0.21	0.04	0.17	0.04	0.04
GBM	0.86	0.73	0.73	0.07	0.73
XGBoost	1.39	1.94	1.03	0.05	1.94
DRF	4.52	20.39	3.34	0.31	20.39
XRT	6.91	47.73	5.03	0.35	47.73

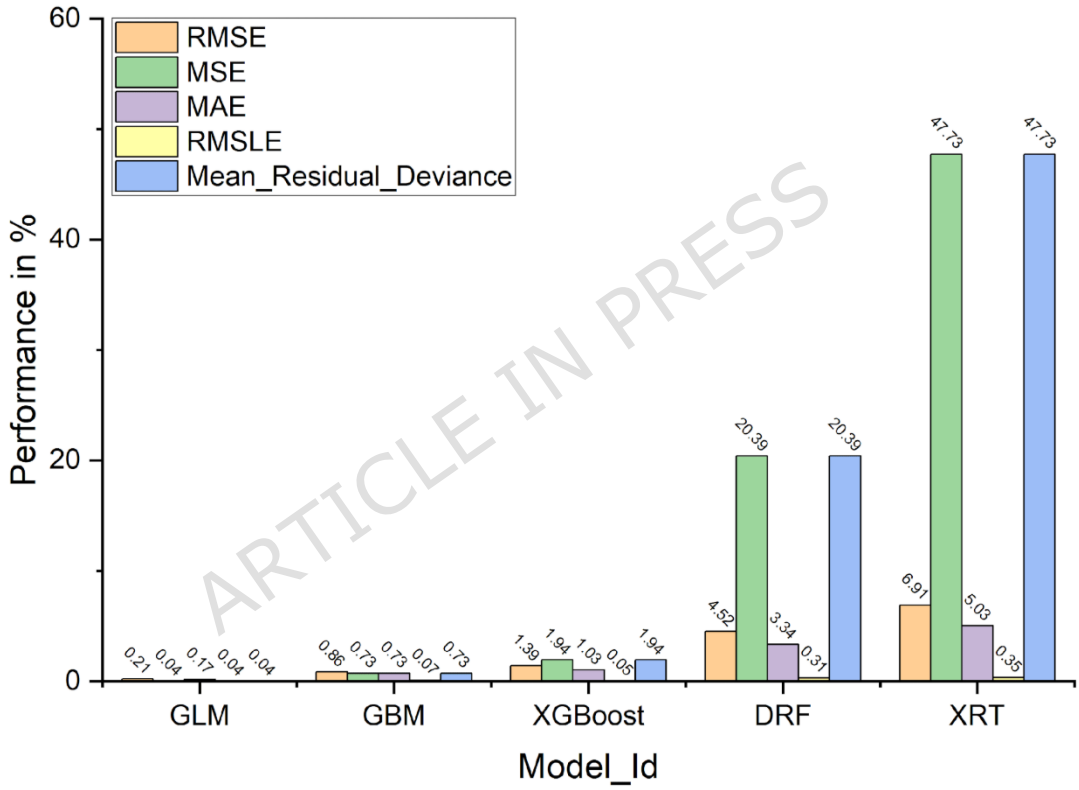

Figure 10. Graphical Plot of Performance Comparison

As shown in the table, the GLM method has the lowest RMSE value with 0.21, representing it as the best model in this comparison. Following the GLM, the GBM has the closest result (0.86), making it the second-best model in RMSE, and the least performed model was XRT with 6.91 RMSE. The GLM model has the lowest MSE value at 0.04 compared to all other models in this comparative analysis. The GLM has the best performance results in all the parameters with lower values, making it one of the best models in this comparative analysis. Compared to the GLM, the GBM has the second-best results overall, and the least performed model was XRT, with relatively higher values in the observed parameters. The GLM model has constant low values on all the performance metrics, representing its

capability in the battery prediction with good performance. However, the GBM and XGBoost also performed better in a few metrics. However, the GLM model has the overall best performance across all the performance parameters in this comparative study.

5. Discussion and Limitations

This comparative study demonstrates that SOC prediction performance is highly dependent on dataset characteristics, feature selection, and model complexity. While GLM exhibited strong performance due to linear feature–SOC relationships, ensemble models showed signs of overfitting and sensitivity to hyperparameters. The absence of cross-validation across diverse driving cycles limits claims of universal generalization. Additionally, the study relies on a single public dataset obtained under controlled laboratory conditions, which may not fully represent real-world EV operating environments. Future work should incorporate multi-dataset validation, domain-guided feature engineering such as incremental capacity metrics, and robustness testing across dynamic drive cycles to enhance generalizability.

6. Conclusion

This work presented a comparative evaluation of five supervised machine learning models for SOC prediction of lithium-ion batteries using the SiCWell dataset. Rather than identifying a universally optimal model, the results demonstrate that model performance strongly depends on feature linearity, dataset scale, and operating conditions. The GLM model achieved low prediction errors under controlled conditions due to strong linear correlations between selected features and SOC. However, ensemble models such as GBM and XGBoost exhibited higher sensitivity to hyperparameter configurations and dataset size. The study emphasizes the importance of transparent dataset usage, interpretability, and cautious generalization when applying ML models in battery management systems. Future research will focus on cross-dataset validation, feature ablation studies, and hybrid physics-informed ML approaches.

Declarations

Ethics Approval and Consent to Participate:

No participation of humans takes place in this implementation process.

Human and Animal Rights:

No violation of Human and Animal Rights is involved.

Funding:

No funding is involved in this work.

Data availability statement:

The datasets used and/or analysed during the current study available from the corresponding author on reasonable request.

Conflict of Interest:

Conflict of Interest is not applicable in this work.

Authorship contributions:

Mandeddu Sudhakar Reddy - Writing- original draft, Project administration, Resources

M. Monisha - Writing- review & editing, Conceptualization, Data Curation

Acknowledgement:

There is no acknowledgement involved in this work.

References

[1] M. Adnene, A. Khomsi, and J. P. Trovao, “Efficient Management of Energy Consumption of Electric Vehicles Using Machine Learning—A Systematic and Comprehensive Survey,” Energies, vol. 16, 4897, 2023.

[2] W. Liu, T. Placke, and K. T. Chau, “Overview of batteries and battery management for electric vehicles,” Energy Reports, vol. 8, pp. 4058–4084, 2022.

[3] A. Kamul, N. Hananda, C. Harito, E. Djuana, G. N. Elwirehardja, B. Pardamean, F. E. Gunawan, A. S. Budiman, M. Asrol, A. A. N. P. Redi, and T. Pasang, “Machine Learning Application in Battery Prediction: A Systematic Literature Review and Bibliometric Study,” E3S Web of Conferences, vol. 388, 01021, 2023.

[4] H. Rauf, M. Khalid, and N. Arshad, “Machine learning in state of health and remaining useful life estimation: Theoretical and technological development in battery degradation modelling,” Renewable and Sustainable Energy Reviews, vol. 156, 111903, 2022.

[5] Z. Ren and C. Du, “A review of machine learning state-of-charge and state-of-health estimation algorithms for lithium-ion batteries,” Energy Reports, vol. 9, pp. 2993–3021, 2023.

[6] M. S. Lippu, M. A. Hanan, T. F. Karem, A. Husain, M. H. Saad, A. Ayyob, M. S. Miaah, T. M. I. Mahalia, “Intelligent algorithms and control strategies for battery management system in electric vehicles: Progress, challenges and future outlook,” Journals of Cleaner Productions, vol. 292, 126044, 2021.

[7] P. Sharma and B. J. Bora, “A Review of Modern Machine Learning Techniques in the Prediction of Remaining Useful Life of Lithium-Ion Batteries,” Batteries, vol. 9, 13, 2023.

[8] M. F. Nag, J. Zao, Q. Yang, G. J. Condut, and Z. W. Sheh, “Predicting the state of charge and health of batteries using data-driven machine learning,” Nature Machines Intelligences, vol. 2, pp. 161–170, 2020.

[9] A. A. Miaari and H. M. Ali, “Batteries temperature prediction and thermal management using machine learning: An overview,” Energy Reports, vol. 10, pp. 2277–2305, 2023.

[10] C. Vidale, P. Malysiz, P. Kolmeyer, and A. Emaidi, “Machine Learning Applied to Electrified Vehicle Battery State of Charge and State of Health Estimation: State-of-the-Art,” IEEE Access, vol. 9, pp. 52795-52816, 2020.

[11] X. Liu, D. Yu, V. S. Byag, and D. I. Stroe, “The development of machine learning-based remaining useful life prediction for lithium-ion batteries,” Journals of Energy Chemistry, vol. 83, pp. 104–122, 2023.

[12] S. Jinn, X. Sui, X. Hang, S. Zang, R. Teodoresu, and D. I. Store, “Overview of Machine Learning Methods for Lithium-Ion Battery Remaining Useful Lifetime Prediction,” Electronic, vol. 11, 3126, 2021.

[13] I. Ulah, K. Liu, T. Yamammoto, R. E. A. Mamlok, and A. Jamaal, “A comparative performance of machine learning algorithm to predict electric vehicles energy consumption: A path towards sustainability,” Energy & Environment, vol. 33, no. 8, pp. 1583–1612, 2022.

[14] S. Sun, J. Zang, J. Bi, and Y. Zhang, “A Machine Learning Method for Predicting Driving Range of Battery Electric Vehicles,” Journals of Advanced Transportations, vol. 19, pp. 1-15, 2019.

[15] H. Rauf, M. Khalid, and N. Arshad, “A novel smart feature selection strategy of lithium-ion battery degradation modelling for electric vehicles based on modern machine learning algorithms,” Journals of Energy Storages, vol. 68, 107577, 2023.

[16] K. Das, R. Kumar, and A. Krishna, “Analyzing electric vehicle battery health performance using supervised machine learning,” Renewable and Sustainable Energy Reviews, vol. 189, 113967, 2024.

[17] Z. Fei, F. Yang, K. L. Tsui, L. Li, and Z. Zang, “Early prediction of battery lifetime via a machine learning based framework,” Energy, vol. 225, 120205, 2021.

[18] S. Jafari and Y. C. Byun, “Efficient state of charge estimation in electric vehicles batteries based on the extra tree regressor: A data-driven approach,” Heliyon, vol. 10, e25949, 2024.

[19] B. Zhang and G. Ren, “Li-Ion Battery State of Charge Prediction for Electric Vehicles Based on Improved Regularized Extreme Learning Machine,” World Electrical Vehicles Journal, vol. 14, 202, 2023.

[20] S. Jafari, Z. Shahbazi, and Y. C. Byun, “Lithium-Ion Battery Health Prediction on Hybrid Vehicles Using Machine Learning Approach,” Energies, vol. 15, 4753, 2022.

[21] L. Zhao, W. Yao, Y. Wang, and J. Hu, “Machine Learning-Based Method for Remaining Range Prediction of Electric Vehicles,” IEEE Access, vol. 8, pp. 212423-212441, 2020.

[22] M. H. Sulaiman and Z. Mustaffa, “State of Charge Estimation for Electric Vehicles Using Random Forest,” Green Energies and Intelligent Transportations, 100177, 2024.

[23] V. Chandhran, C. K. Patel, A. Karthik, D. Ganeshperumal, R. Rahem, and A. Gosh, “State of Charge Estimation of Lithium-Ion Battery for Electric Vehicles Using Machine Learning Algorithms,” World Electric Vehicle Journal, vol. 12, 38, 2021.

[24] D. N. T. How, M. A. Hanan, M. S. Lipu, K. S. M. Sahar, P. J. Kerr, and K. M. Mutaqi, “State-of-Charge Estimation of Li-Ion Battery in Electric Vehicles: A Deep Neural Network Approach,” IEEE Transactions on Industry Applications, vol. 56, no. 5, pp. 5565-5574, 2020.

[25] F. Marchiori and M. Conti, “Your battery is a blast! safeguarding against counterfeit batteries with authentication,” In Proceeding of the ACMSIGSAC Conferences on Computers and Communication Security, 2023, pp. 105-119.

[26] A. Guissan, T. C. Edwards, and T. Hasstie, “Generalized linear and generalized additive models in studies of species distributions: setting the scene,” Ecological Modeling, vol. 157, no. 2-3, pp. 89-100, 2002.

[27] B. Esmaeili, M. R. Hallowell, and B. Rajagopalan, “Attribute-Based Safety Risk Assessment II: Predicting Safety Outcomes Using Generalized Linear Models,” Journal of construction engineering and management, vol. 141, no. 8, 04015022, 2015.

[28] R. Sibindi, R. W. Mwangi, and A. G. Waititu, “A boosting ensemble learning based hybrid light gradient boosting machine and extreme gradient boosting model for predicting house prices,” Engineering Reports, vol. 5, no. 4, e12599, 2023.

[29] G. W. Cha, H. J. Moon, and Y. C. Kim, “Comparison of Random Forest and Gradient Boosting Machine Models for Predicting Demolition Waste Based on Small Datasets and Categorical Variables,” International Journal of Environmental Research and Public Health, vol. 18, no. 16, 8530, 2021.

[30] M. T. Anwar, E. Winarno, W. Hadikurniawati, and M. Novita, “Rainfall prediction using Extreme Gradient Boosting,” Journal of Physics: Conference Series, vol. 1869, no. 1, 012078, 2021.

[31] Z. A. Ali, Z. H. Abduljabar, H. A. Taheer, A. B. Salow, and S. M. Almuffti, “Exploring the Power of eXtreme Gradient Boosting Algorithm in Machine Learning: A Review,” Academic Journals of Nawroz University, vol. 13, no. 3, pp. 321-334, 2023.

[32] O. Alshboul, A. Shehadeh, G. Almasabha, and A. S. Almuflih, “Extreme Gradient Boosting Based Machine Learning Approach for Green Building Cost Prediction,” Sustainability, 14(11), 6651, 2022.

[33] M. Asgari, W. Yang, and M. Farnaghi, “Spatiotemporal data partitioning for distributed random forest algorithm: Air quality prediction using imbalanced big spatiotemporal data on spark distributed framework,” Environmental Technology & Innovation, vol. 27, 102776, 2022.

[34] L. K. Shrivastav and R. Kumar, “An Ensemble of Random Forest Gradient Boosting Machine and Deep Learning Methods for Stock Price Prediction,” Journals of Informational Technology Research, vol. 15, no. 1, pp. 1-19, 2022.

[35] H. Khwon, J. Parkk, and Y. Lei, “Stacking Ensemble Technique for Classifying Breast Cancer,” Healthcare Informatic Research, vol. 24, no. 3, pp. 282-288, 2019.

[36] H. A. Najada and I. Mahgoub, “Real-Time Incident Clearance Time Prediction Using Traffic Data from Internet of Mobility Sensors,” 2017 IEEE 15th Intl Conf on Dependable, Autonomic and Secure Computing, 15th Intl Conf on Pervasive Intelligence and Computing, 3rd Intl Conf on Big Data Intelligence and Computing and Cyber Science and Technology Congress (DASC/PiCom/DataCom/CyberSciTech), 2017, pp. 728-735.

[37] Y. Xi, C. Zhui, R. Hu, and Z. Zu, “A Coarse-to-Fine Approach for Intelligent Logging Lithology Identification with Extremely Randomized Trees,” Mathematical Geoscience, vol. 54, pp. 858-877, 2021.

[38] M. T. Udin and M. A. Uddiny, “Human Activity Recognition from Wearable Sensors using Extremely Randomized Trees,” In 2015 International conferences on electrical engineering and information communications technology, 2015, pp. 1-6.

[39] P. Kumar, S. N. Singh, and S. Dawra, “Software Component Reusability Prediction Using Extra Tree Classifier and Enhanced Harris Hawks Optimization Algorithm,” International Journals of Systems Assurance Engineering and Managements, vol. 13, no. 2, pp. 892-903, 2022.

[40] J. N. Sekhar, B. Domathoti, and E. D. Santibanez Gonzalez, “Prediction of Battery Remaining Useful Life Using Machine Learning Algorithms,” Sustainability, vol. 15, 15283, 2023.

[41] Ma, K., Yang, J., & Liu, P. (2020). Relaying-Assisted Communications for Demand Response in Smart Grid: Cost Modeling, Game Strategies, and Algorithms. IEEE Journal on Selected Areas in Communications, 38(1), 48-60. doi: 10.1109/JSAC.2019.2951972

[42] Wang, M., Zhou, D., & Chen, M. (2023). Hybrid variable monitoring: An unsupervised process monitoring framework with binary and continuous variables. Automatica, 147, 110670. doi: https://doi.org/10.1016/j.automatica.2022.110670

[43] Kuang, H., Liao, H., Zhang, Z., Huang, S., Mao, B., Shan, W.,... Li, T. (2026). Dynamic diels-alder reaction crosslinked metal-organic framework/poly (ionic liquid) composite solid electrolyte for lithium-metal batteries. Journal of Colloid and Interface Science, 707, 139638. doi: https://doi.org/10.1016/j.jcis.2025.139638

[44] Hao, W., Guo, F., Li, J., & Xie, J. (2026). Influence of Physical Parameters on Lithium Dendrite Growth Based on Phase Field Theory. Metals, 16(1), 41. https://doi.org/10.3390/met16010041

[45] Long, T., & Guo, Y. (2025). Research on the temperature radius stratification model based on electrochemical-thermal-force coupling in Lithium-ion batteries. Electrochemistry Communications, 180, 108052. doi: https://doi.org/10.1016/j.elecom.2025.108052

[46] Li FJ, Zhang HY, Lu ZL et al. Unsupervised learning enabled label-free single-pixel imaging for resilient information transmission through unknown dynamic scattering media. Opto-Electron Adv 8, 250013 (2025). DOI: 10.29026/oea.2025.250013